



UNIVERSIDADE FEDERAL DO MARANHÃO
COORDENAÇÃO DO CURSO DE ENGENHARIA DA COMPUTAÇÃO/CCET
SISTEMAS DISTRIBUÍDOS

ESTRUTURA ANALITICA DO PROJETO (EAP)

DISCENTES:

FRANCISCO GABRIEL SANTOS - 2020014544
KEVEN GUSTAVO DOS SANTOS GOMES - 2020034420
KAUAN GARCIA PEREIRA MARTINS - 2021026595
JOÃO PEDRO MIRANDA SOUSA - 2022011087
WESLEY DOS SANTOS GATINHO - 2020051056

PROFESSOR: LUIZ HENRIQUE NEVES RODRIGUES

1 INTRODUÇÃO

Este documento apresenta a Estrutura Analítica do Projeto (EAP) para o desenvolvimento do "Sistema Distribuído de Gerenciamento de Tarefas com Comunicação Publish-Subscribe", realizado no contexto da disciplina de Sistemas Distribuídos da Universidade Federal do Maranhão. A EAP detalha os principais componentes do planejamento do projeto, incluindo o escopo detalhado, o cronograma de execução, a estimativa de custos envolvidos e uma análise dos principais riscos identificados. O objetivo é fornecer uma visão clara e estruturada do trabalho a ser realizado, servindo como guia para a equipe de desenvolvimento e para o acompanhamento do progresso pelo professor orientador. O projeto baseia-se no Termo de Abertura de Projeto (TAP) previamente elaborado pela equipe.

1 ESCOPO DO PROJETO

1. Backend:
 - API REST (CRUD de tarefas).
 - Middleware - Broker RabbitMQ para eventos (ex.: "tarefa_editada").
2. Segurança:
 - Autenticação JWT.
3. Tolerância a Falhas:
 - Réplica primária-secundária do servidor.

2 RECURSOS NECESSÁRIOS

2.1 Software (SW):

- Visual Studio Code (ambiente de desenvolvimento)
- Docker (ambiente de containers)
- Postman (testes de API REST)

- Git e GitHub (controle de versão e colaboração)
- RabbitMQ (sistema de mensageria Publish-Subscribe)
- MySQL ou PostgreSQL (banco de dados relacional)

2.2 Hardware (HW):

- Computadores pessoais dos discentes com acesso à internet
- Servidor local ou em nuvem para testes (ex.: máquina virtual com Linux)

2.3 Recursos Humanos (RH):

- Cinco discentes para desenvolvimento, testes e documentação
- Professor orientador para supervisão técnica e validação
- Usuários simulados ou reais para testes e validações de usabilidade

3 RISCOS

Risco	Probab.	Impacto	Estratégia de Resposta
Dificuldade com integração RabbitMQ	Média	Alta	Treinamento prévio, prototipação
Falhas de sincronização offline	Média	Alta	Testes extensivos, logs detalhados
Problemas de configuração TLS/JWT	Baixa	Média	Uso de tutoriais, revisão por pares
Atraso devido à sobrecarga acadêmica	Alta	Média	Planejamento realista, divisão tarefas
Falta de engajamento dos usuários	Média	Baixa	Comunicação clara, testes iterativos

4 CRONOGRAMA

Cronograma Detalhado do Projeto - Sistema Distribuído de Gerenciamento de Tarefas

Concluído (antes de hoje)Em ExecuçãoPlanejadoFuturo

Tarefas	06/04	20/04	04/05	18/05	28/05	15/06	29/06	13/07	27/07
TAP - PLANEJAMENTO									
1.1 Definição Detalhada do Escopo e Requisitos									
1.2 Escolha Final das Tecnologias									
1.3 Desenho da Arquitetura Detalhada									
1.4 Configuração do Ambiente de Dev									
EXECUÇÃO									
2.1 Implementação API REST (CRUD Básico)									
2.2 Modelagem e Configuração BD									
2.3 Implementação Serviço RPC/gRPC									
2.4 Configuração Event Broker (RabbitMQ/MQTT)									
3.1 Integração Broker (Publish)									
3.2 Implementação Automática (MQTT/TLS)									
3.3 Dev microserviços Others									
3.4 Implementação Teletônica e Python									
MONITORAMENTO									
4.1 Testes Unitários e Integração									
ENTREGA									
4.2 Elaboração Documentação Técnica									
4.3 Testes Segurança e Terceiros Finais									
4.3 Testes Encurtadores									
4.4 Preparação Apresentação Final									
4.6 Entrega Final / Apresentação									

Data atual: 26 de Maio de 2025 | Linha vermelha indica a data atual